

Gerő Péter

Budapest

A mesterséges intelligencia alkalmazása

Számítástechnika tanulása felnőtt korban

Egy folyamatban lévő projekt első eredményei alapján mutatunk be két, esetleg általánosítható mérőeszközt a számítástechnikai feladatmegoldás elemzésére, illetve modellezésére mesterséges intelligencia segítségével.

Lassan felnő az első olyan nemzedék, amelynek a tagjai gyermekkorukban tömegesen találkozhattak a számítógéppel. Ez új helyzetet teremt a számítástechnika-tanulásban, és nemcsak a gyermekek esetében: már a fiatal felnőttek képzésében is zömmel olyan tanulókra számíthatunk, akiknek vannak számítástechnikai előismereteik.

A tapasztalat azonban azt mutatja, hogy ebből a nemzedékből sem mindannyian kerültek gyermekkorukban érdemi kapcsolatba a számítógéppel (sokszor azok sem, akiknek erre lehetőségük lett volna). Akik igen, azok közül is sokaknak csak néhány kezelési alap-sémára terjednek ki az ismereteik, s ezek az ismeretek néha meglehetősen hiányos alapokra épülnek és rendszerezetlenek. Akik pedig ezen túlmenő számítástechnikai ismeretekkel, illetve áttekintéssel kezdik a felnőtt életüket, azoknak is szükségük van további alapos képzésre, amint abba a helyzetbe kerülnek, hogy a számítógépet problémamegoldásra, kutatásra, önálló eredmények elérésére használják.

A felnőttkori számítástechnika-tanulás tehát egyáltalán nem vesztette el az aktualitását, mi több: a folyamat egyre fontosabbá és módszertanilag egyre kényesebbé válik. Ez önmagában is érdekessé és aktuálissá teszi a számítástechnika-tanulás sikerességének tényezőire vonatkozó kutatást.

A szakirodalom tanulmányozása ugyanakkor arra vezet, hogy az álláspontok nem egységesek abban a tekintetben, mit nevezhetünk számítástechnikai tudásnak. Ez azt jelenti, hogy nehéz alapot találni ahhoz, hogy a tanulás sikerességét vizsgálhassuk.

Mit tudjunk – s honnan tudható, hogy azt tudjuk-e?

A társadalmi gyakorlat szükséglete nyilvánvalóan az értelmes, használható tudás. A gyakorlatban azonban a tudás mérése alig léphet túl a lexikális ismeretek valamiféle viszszafejtésének és a feladatmegoldási sémák alkalmazásának a szintjén. Ha ugyanis a tanulót olyan helyzetbe hozzuk, hogy a tanultakat valóban használja, rögtön abba a ténybe ütközünk, hogy az egyes embereknek nemcsak a konkrét gondolati lépéseik, hanem még a kiinduló stratégiájuk is alapvetően eltérő lehet egy-egy feladat megoldásakor. Amikor tehát különböző személyeket figyelünk meg ugyanazon feladat megoldása közben, akkor egyáltalán nem biztos, hogy szubjektíve is ugyanazon a feladaton dolgoznak.

Figyelemre méltó tanulságokkal jár ebben a tekintetben a közvetlen transzlációs és a modell-alkotó gondolkodás szembeállítás. (Mayer és Hegarty, 1998)

A közvetlen transzlációs stratégiának az a jellegzetessége, hogy számokat és kulcsszavakat keresünk a (szöveges) feladatban és ezekkel aritmetikai műveleteket készítünk elő.

A másik lehetséges stratégia a problémamodellezés: megértjük a problémában leírt helyzetet, és a szituáció reprezentációján alapuló megoldási tervet készítünk.

Figyeljünk meg (az említett tanulmányban közöltek nyomán) egy-egy példát, amelyben a két stratégia a megfigyelő számára megkülönböztethetetlen, illetve megkülönböztethető:

1. példa

Mariska néni boltjában 1 liter tartós tej 26 forinttal drágább, mint Juliska néni boltjában, ahol 229 forint. Mennyibe kerül 3 liter tej Mariska néni boltjában?

közvetlen translációs megoldás:

kiinduló ár: 229 Ft/l; különbség: 26 Ft/l

a „drágább” szó összeadásra utal!

tehát az ár: $229 \text{ Ft/l} + 26 \text{ Ft/l} = 255 \text{ Ft/l}$

$3 \text{ l} \times 255 \text{ Ft/l} = 765 \text{ Ft}$

problémamodellező megoldás:

Juliska néninél: 229 Ft/l; különbség: 26 Ft/l

Mariska néninél = Juliska néninél + 26 Ft/l

tehát Mariska néninél $229 \text{ Ft/l} + 26 \text{ Ft/l} = 255 \text{ Ft/l}$

$3 \text{ l} \times 255 \text{ Ft/l} = 765 \text{ Ft}$

A gondolkodási folyamat alapvetően más, az eredmény mégis ugyanaz. Egy iskolai dolgozatban például (ha a tanuló le nem írja lépésről lépésre, hogy hogyan gondolkodott) a két stratégia nem különböztethető meg egymástól. Lássuk most ugyanennek a szöveges példának egy másik, alig átfogalmazott változatát.

2. példa

Mariska néni boltjában 1 liter tartós tej 255 forint: ez 26 forinttal drágább, mint Juliska néni boltjában. Mennyibe kerül 3 liter tej Juliska néni boltjában?

közvetlen translációs megoldás:

kiinduló ár: 255 Ft/l; különbség: 26 Ft/l

a „drágább” szó összeadásra utal!

tehát az ár: $255 \text{ Ft/l} + 26 \text{ Ft/l} = 281 \text{ Ft/l}$

$3 \text{ l} \times 281 \text{ Ft/l} = 843 \text{ Ft}$

problémamodellező megoldás:

Mariska néninél: 255 Ft/l; különbség: 26 Ft/l

Juliska néninél = Mariska néninél – 26 Ft/l

tehát Juliska néninél $255 \text{ Ft/l} - 26 \text{ Ft/l} = 229 \text{ Ft/l}$

$3 \text{ l} \times 229 \text{ Ft/l} = 687 \text{ Ft}$

A közvetlen translációs stratégia lényege tehát az, hogy a szövegből kiragadjuk az adatokat, s a kulcsszavakból következtetünk a végrehajtandó műveletre. Nyilvánvaló, hogy ez a stratégia felületesebb, és szélsőséges esetben akár rendkívüli mértékű tévedésekhez is vezethet. Ugyanakkor a kétféle gondolkodásmód közötti különbség jobbra csak „erre kialakított” feladatok esetében válik láthatóvá – márpedig a matematikai feladatgyűjtemények legtöbb szöveges feladata közvetlen translációs stratégiával is megoldható. (*Briars és Larkin*, 1984) Ha figyelembe vesszük, hogy a problémamodellező gondolkodás semmiképpen sem gyorsabb a közvetlen translációséknál, akkor akár azt is kimondhatjuk, hogy az iskola az esetek többségében a közvetlen translációs stratégiát jutalmazza!

Tanulmányunk keretein messze túlmegy annak az elemzése, hogy amit tudás-transzfer néven gyakran hiányolunk, az nem a problémamodellezés hiánya-e; hogy a kétféle stratégia hogyan és mitől alakul ki a tanulóknál, illetve hogy egyes helyzetekben hogyan és mitől választják egyiket vagy a másikat. Hogyan lehetne a problémamodellezést a jelenleginél szisztematikusabb módon tanítani, fejleszteni? A metakognitív gondolkodási stratégiák fejlesztésére jobbra csak egyedi, óvatos, minden részletükben nem is magyarázható kísérleteket ismerünk: egy ilyenről, s az annak alapján tervezett hazai kísérletről az Iskolakultúra is szólt már. (*Csikós*, 2003)

A gondolat nagymértékben súlyosbíthatja, ha a feladaton dolgozó személy gondolkodásába beépült az az igen elterjedt tévhit, hogy „minden problémának csak egy »helyes« megoldása van. (*Reusser és Stebler*, 1997) Ilyenkor ugyanis a tanuló erőfeszítése arra irányul, hogy visszaemlékezzen a „megoldási sémára” – mert fel sem merül benne, hogy más módszerrel is (ugyanazon vagy akár eltérő) helyes eredményre juthat. A fentebbi idézetre hivatkozó hazai cikk kimutatja, hogy amint valaki úgy érzi, hogy „rátalált a helyes vágányra”, törtrészére csökken a „realisztikus” megoldás megtalálásának esélye.

Ami mindezekből témánkhoz kötődik: úgy sejthető, hogy a számítástechnikában, ha lennének hasonló vizsgálatok, azok nagyon is hasonló feladatmegoldó gondolatmeneteket fednének fel. Nincs okunk feltételezni, hogy ne nyílna itt is tér a közvetlen translációs gondolkodás számára; és sokféle követelményrendszer létezik, amely nem is vár ennél többet.

Ahhoz, hogy az eltérő feladat-értelmezést, eltérő stratégiát és gondolatmenetet alkalmazó személyek között különbséget tehessünk, különlegesen megfogalmazott feladatokra vagy különleges mérési helyzetre van szükség.

Súlyos gond, hogy az eredményen gyakran nem látszik, milyen gondolkodás állt mögötte. A téves eredmény lehet jó gondolatmenet alkalmazása mellett elkövetett elírás vagy számolási hiba hatása is, miközben a jó eredmény forrása „ráhibázás” vagy véletlen egyezés is lehet. Ennek az esélye egy számítógép-használói feladat esetében sokkal nagyobb, mint például a matematikában. Nyilvánvalóan nem tehetjük meg, hogy minden egyes diák mellé jegyzetelő megfigyelőt állítunk: ennek munka-igénye erősen behatárolná vizsgálataink hosszát és létszámát.

Egy ötlet az adott körülmények ügyes felhasználására

Ha számítógépes feladatmegoldásról beszélünk, akkor egy különleges, más szakterületen alig vagy egyáltalán nem alkalmazható lehetőségünk adódik. Ez nem más, mint hogy a feladatmegoldás folyamatát is gépi úton rögzíthetjük és utólag precízen elemezhetjük!

Ha tehát olyan számítógépes feladatot hozunk létre, amely a kétféle stratégiát ügyesen eltérő utakra tereli, akkor a megoldás folyamatát (a vizsgált személyek tudtával!) naplózva ebben a naplóban utólag megkereshetjük az egyik, illetve a másik stratégiára utaló gondolkodás jeleit. Módunk van tehát arra, hogy felismerjük a jó gondolatmenetet akkor is, ha a vizsgált személy a feladatmegoldás folyamatában valahol hibát követett el, vagy ha nem ért a folyamat végére; ugyanakkor kiszűrhetjük a ráhibázásos jó eredményeket.

Ennek a feladatnak olyannak kell lennie, amely nem igényel igazán sok előismeretet (azaz a vizsgált személyek között ne legyen nagy szórás abban a tekintetben, hogy milyen mértékig vannak a szükséges előfeltételek birtokában). A legjobb az olyan feladat, amelynek megoldásához a szükséges előismeretek annyira csekélyek és annyira egyszerűek, hogy bármely ép ember rövid idő alatt elsajátíthatja őket – s ekkor a mérést egy képzéssel vagy szintre hozó ismétléssel kezdhethetjük Ettől kezdve feltételezhető (ha kell: tesztel bizonyítható), hogy a vizsgált személyek ebből a szempontból azonos szinten vannak. Ez azt jelenti, hogy ha a szükséges előismereteket úgy írjuk le (és olyan stílusban), hogy csak a legáltalánosabb, mindenkiről feltételezhető előzetes tudás- és készség-szintet igényeljük, ez sem lehet 4–5 oldalnál több (esetleges szemléltetéseket, ábrákat leszámítva). Ennek az előismeretnek ugyanakkor olyannak kellene lennie, hogy aki nincs a birtokában, annak számára minél kevesebb „ráhibázási” lehetőséget adjon. Jellegetesen ilyenek például az egyedi, szokás alapján szabvánnyá váló jelölések: aki nem ismeri őket, kitalálni sem tudhatja. Különlegesen előnyös lehet, ha olyan feladatot fogalmazunk meg, amelynek az előismeretei iskolán, erre irányuló képzésen, célzott tanuláson kívüli helyzetben nem sajátíthatóak el: ekkor, későbbi vizsgálatban még arra is lehetőségünk nyílik, hogy magának a képzésnek a hatását mérjük, mert feltételezhető, hogy ha az adott populáció az adott (vagy más hasonló) képzésben nem vett részt, akkor az adott előismerettel nem rendelkezik.

Ilyen célra használom az alábbi Excel-feladatot.

Oldja meg az alábbi feladatot úgy, hogy csak egyetlen formulát kelljen begépelni, a többi másolással állíthassa elő:

Egy gyártmány ötféle technológiával állítható elő. Az elsőhöz 10 gép kell és 15 ember. A másodikhoz 8 gép is elég, de 25 ember kell. A harmadikhoz, negyedikhez, ill. ötödikhez szükséges gépek száma rendre 6, 4, ill. 2, a szükséges emberek száma pedig 33, 40, ill. 45.

Háromféle gép létezik. Bármelyik gép bármelyik technológiához használható. A gépek darabonkénti és óránkénti bérleti díja 2.000, 1.600, ill. 1.440 Ft. Minél egyszerűbb a gép, annál nehezebb az emberi munka. Az emberek az első gép mellett 400, a második mellett 500, a harmadik mellett 560 Ft órabérért dolgoznak.

Számolja ki a technológiák, illetve gépek szerinti óránkénti ráfordítást!

Az óránkénti ráfordítás az embereknek kifizetett órabérek és a gépbérleti díjak összege.

Az embereknek kifizetett órabérek: az egy főre jutó órabér szorozva az emberek számával.

A gépbérleti díjak: az egy gépre jutó bérleti díj szorozva a gépek számával.

A feladat megoldását 15 formula jelenti (5 technológia, technológiánként 3–3 gép).

A feladat megoldásának célszerű módja:

- a kezdőadatokat begépelése és az eredmény-adatok helyének kijelölése (például az alábbi minta szerint, ahol a megoldást jelentő formuláknak a kérdőjelek helyére kell kerülniük – de természetesen másképp is lehet);
- egy formula létrehozása (például> I. sz. technológia, 1. sz. gép: az alábbi mintán a B14 cella tartalma);
- a formula átmásolása a többi 14 helyre.

	A	B	C	D	E	F	G	H
1	Excel-feladat							
2								
3	techn.:	I.	II.	III.	IV.	V.		
4	gép	10	8	6	4	2		
5	ember	15	25	33	40	45		
6								
7		bérl. díj	órabér					
8	gép: 1.	2 000	400					
9	gép: 2.	1 600	500					
10	gép: 3.	1 440	560					
11								
12	Óránkénti ráfordítás:							
13		I.	II.	III.	IV.	V.		
14	gép: 1.	?	?	?	?	?		
15	gép: 2.	?	?	?	?	?		
16	gép: 3.	?	?	?	?	?		

1. ábra

A feladat tehát – ha az instrukciót követjük – abból áll, hogy a bal felső kérdőjel helyére (a B14 cellába) olyan Excel-formulát kell írni, amely:

- megoldja a feladatot az 1. gép és az I. technológia esetére;
- ha pedig a B14:F16 tartomány bármely más cellájába másoljuk, akkor ott is az adott cellába illő formula fog megjelenni.

Az egyetlen helyes megoldás ez a formula: =B\$4*\$B8+B\$5*\$C8 (ahol az összeadások természetesen felcserélhetők és az összeadásban szereplő szorzások tényezőinek is tetszőleges lehet az egymás közti sorrendje).

A formula egyáltalán nem áttekinthetetlenül hosszú vagy bonyolult. Ugyanakkor, ha valaki nem tanulta meg a hozzá való előismereteket, akkor magától nem jöhet rá: a relatív és az abszolút hivatkozást az emberi gondolkodás használja ugyan, de explicit módon nemigen fejezi ki, a \$-jelölés pedig végképp kitalálhatatlan.

Gondoljuk meg: ez a feladat a korábban felsorolt valamennyi feltételnek megfelel – egyetlen, ki nem mondott feltételtől eltekintve, hogy tudniillik érdekes, figyelemfelkeltő, stimuláló legyen...

A feladat „trükkje” az, hogy a megoldáshoz olyan formulát kell létrehozni, amely a relatív és az abszolút cellahivatkozásokat vegyesen használja – miközben a tankönyvek és Excel-leírások közül egyet sem ismerek, amelyik ezt a vegyes alkalmazást hangsúlyosan tanítaná, és ilyen tanfolyammal sem találkoztam még. Ha tehát a tanuló (akinek, tanuló voltából következően, nem lehet még annyi rutinja és magabiztossága, hogy kellő rálátással szemlélje a témakört, és „zsonglörködni” tudjon a megtanult jelölés-elemekkel) az alkalmazható sémák megtalálása útján akarja megoldani a feladatot, akkor olyan formulák alkalmazásával kísérletezik, amelyek esetében minden egyes cellahivatkozás vagy a sor- és az oszlop-hivatkozás is abszolút, vagy egyik sem az.

Az elemzés ezután annyiból áll, hogy a gépi napló alapján felismerjük egyrészt a séma-alkalmazás, illetve a találgatás, másrészt a szisztematikus feladatmegoldás jeleit.

Ez még mindig hosszadalmas. (Eddigi vizsgálataimban, ahol a gép-napló elkészítésére a Macro Magic 4.1 programot használtam,

*Amire nagyon vigyázni kell:
ahogy tudomásul vesszük, hogy
a gép által követett
feladatmegoldási folyamat nem
az emberi gondolkodást utánoz-
za, ugyanúgy nem szabad felté-
teleznünk ennek a fordítottját
sem. A Prolog sajátosságai és a
feladat gépi megfogalmazásá-
nak módja miatt a jelen esetben
a program az összes lehetséges
megoldási gondolatmenetet
megmutatja; de fel sem merül-
het, hogy az emberek feladat-
megoldó képességét aszerint osz-
tályozzuk (a szó bármely értel-
mében), hogy a megoldás
hosszabb vagy rövidebb, a gép
által előbb vagy később megta-
lált, gépi szempontból egysze-
rűbb vagy bonyolultabb lépés-so-
rozat mintájára született-e.*

re a Macro Magic 4.1 programot használtam, egy 15–20 perces feladatmegoldó munka „naplója” 40 000–200 000 bájttal az illető által végzett műveletek mennyiségétől és milyen-ségétől függően.) Itt azonban segítségül hívhatjuk a mesterséges intelligenciát!

A cél: megtalálni a helyes, illetve a hibás gondolatmenetek felismerésére utaló jellegzetes viselkedés-mintázatokat. A mesterséges intelligencia-alkalmazások pedig általában éppen a mintázatok keresésén, a minta-illesztéseken alapulnak.

A kérdés már „csak” annyi, hogy melyek azok a tevékenység-mintázatok, amelyekből séma-alkalmazó, s melyek azok, amelyekből találgató viselkedésre következtethetünk.

Az első, beprogramozni tervezett „minták” ilyenek: ha a vizsgált személy ugyanazon formulában több javítást végez egymás után; ha egy formulát többször kitorlól és teljes egészében újraír; ha egy formulát más cellába másol, majd a lemásolt formulát javítja; ha egy formulát más cellába másol, majd az eredeti formulát javítja. A mi konkrét példánk esetében jellegzetes tevékenység bármely formulában az abszolút hivatkozás

relatívvá alakítása és ennek a fordítottja is, főképp, ha többszöri oda-vissza alakítással találkozunk.

Ezeket a tevékenység-mintázatokat akkor is fel kell tudni ismerni, ha egyes elemeik között más tevékenységek is vannak. Például: ha egy formula kitorlélése és újraírása között a vizsgált személy a munkalap kiinduló adatain változtat; ha bármiféle formázást végez; ha egér-mozdulatokkal vagy billentyűzettel navigál a képernyőn. Ha a vizsgált személy egyéb szoftver-használatot végez (más programot elindít vagy bezár, ablakot vált és ott bármit tesz), ezt mind ki kell hagyni az elemzésből a feladatmegoldást tartalmazó munkalapra való visszatérésig. Ugyanígy ki kell hagyni a tetszőlegesen hosszú várakozásokat – nem mintha azoknak nem volna meg a maguk jelentése, de a jelen kérdésünk szempontjából irrelevánsak.

Önként adódó ötlet, hogy az elemző algoritmusnak „öntanuló”-nak kell lennie: nem megszámlálnia, csak bejelölnie kell a „gyanús” tevékenységeket, s annak alapján, aho-

gyan utóbb a vizsgálatot végző kutató ezeket a bejelöléseket „visszaigazolja” vagy elveti, növekednie, illetve csökkennie kell a valószínűségnek, hogy a következő alkalommal a hasonló mintázatokat az algoritmus megjelöli-e.

Ez a gondolat megerősíti azt az aggodalmat, hogy a vázolt módszer, miközben nagyságrendekkel növeli meg az elemzés sebességét s ezen keresztül teljesítőképességét, nem csökkenti-e a megbízhatóságát: vajon a valószínűségi alapon történő gépi válogatás nem jelenti-e azt, hogy a tévedések veszélye nagyobb, mint ha a válogatást a kutató végezné. A válasz nyilván: „de igen” és „épp ellenkezőleg”. Igen, ahogyan a gépies válogatások és besorolások hibáiról mosolyra fakasztó anekdoták sokasága szól, s tudjuk: ezek közül sok valóban igaz. Ugyanakkor csökken is a tévedések száma a manuális elemzéshez képest – nehéz ugyanis elképzelni, hogy egy kutató akár csak 30–40 vizsgált személy várhatóan néhányezer oldalt kitevő naplóját (vagy az együttvéve sokszor-tíz órát kitevő feladatmegoldási folyamat valamilyen, más módon készített jegyzőkönyvét) szakszerűen és hibátlanul elemzi.

Feladat-e a feladat?

Van ugyanakkor a kutatásnak egy figyelemreméltó „mellékterméke” is, amely önmagában is érdekes tanulsággal szolgál: konstruktív bizonyítást ad egy olyan kérdésre, amelyre a választ ilyen módszer nélkül csak spekulációval adhattuk volna meg, s ez a válasz nem is bizonyítás, csak a valószínűsíthető eredmény kimondása lett volna.

Arról a kérdésről van szó, hogy egy feladat – valóban feladat-e. Ismert az a megkülönböztetés, amely szerint feladatnak azt nevezzük, ahol a megoldás módja ismert, csak alkalmazni kell; problémáról pedig akkor beszélünk, ha a megoldás módját is most kell ki találni, létrehozni. Ha egy adott tevékenységről be akarjuk látni, hogy az nem probléma-, hanem feladatmegoldás, akkor bizonyosnak kell lennünk abban, hogy a kiadott feladat a vizsgált személy által ismert előismeretek és szabályok birtokában megoldható; azok elegendőek; további előismeretek, illetve szabály-ismeret nem kellenek.

Ahhoz, hogy ilyen állítást tehessünk, a teljesség igényével meg kell tudnunk adni a feladat megoldásához szükséges valamennyi előismeretet (és az azok értelmezéséhez, alkalmazásához szükséges valamennyi előismeretet és így tovább). A kutató, aki aligha tud elvonatkoztatni saját előismereteitől és aligha tudja megakadályozni sajátos – s ebben az esetben igencsak előnytelen – gondolkodási motívumainak (különösen: egység-képzés, ok-keresés stb.) önkéntelen működtetését, nemigen lehet biztos abban, hogy valóban teljes eredményre jutott. Talán csak nagyobb számú kutató egymást kontrolláló felsorolása és több próba-mérés adhat némi bizonyosságot. Még nehezebb annak a belátása, hogy amikor a felsorolt (tegyük fel: teljes) előismeret-halmazt használjuk, akkor a vizsgált személy ebből nem állít-e elő magának (helyesen vagy tévesen) újabb előismereteket.

Furcsa, amikor egy aforizma adja egy kutatási eredmény elméleti alapját, de most ez a helyzet áll elő. *Einstein* híres mondása, mely szerint „a számítógép képes kérdésekre válaszolni, de nem tud önálló kérdést feltenni”, frappáns kiindulás a következő gondolat-hoz: fogalmazzuk meg valamiféle gépi formában a kérdéses előismereteket – s ha a gép azok alapján valóban meg tudja adni a választ, akkor ezzel kétség nélkül bizonyítottuk, hogy az előismeret- és szabály-halmaz teljes, és hogy alkalmazásukkal (minden alkotó gondolkodás nélkül) a kérdés megválaszolható: vagyis hogy a feladat valóban feladat.

Ebben *Hanák Dávid* (a Budapesti Műszaki Egyetem Ph.D.-hallgatója) volt a munkatársam: ő készítette el a feladat és az előismeret- és szabály-halmaz gépi reprezentációját.

Ez a gépi reprezentáció Prolog programnyelven készült: ez a programnyelv különösen alkalmas a mesterséges intelligencia területére eső feladatok megfogalmazására. (A korábban említett minta-keresési algoritmus gépi megvalósításának is ez lehet a leginkább kézre álló eszköze.)

A Prolog alap gondolata az, hogy ne az elvégzendő műveleteket soroljuk fel egy számítógépi programban, hanem – tényekkel és következtetési szabályokkal – egy „világot” írjunk le, majd tegyünk fel kérdéseket ezzel a világgal kapcsolatban. A Prolog-tankönyvek jellegzetes példájával élve: írjuk le, hogy Zoltán szülője Péter, Péter szülője Judit, és valamely A akkor nagyszülője B-nek, ha van olyan C, aki szülője B-nek, s aki-nek A szülője. Ezután megkérdezhetjük, hogy az adott világban van-e Zoltánnak nagyszülője, s a Prolog-program azt feleli, hogy igen, van, mégpedig Judit; vagy megkérdezhetjük, hogy az adott világban van-e Péternek nagyszülője, s a Prolog-program azt feleli, hogy a rendelkezésére álló adatok alapján ezt nem tudja. A mi szempontunkból a lényeg az, hogy a kérdésre (létezik-e az adott világban ilyen és ilyen feltételeknek megfelelő elem?) a Prolog-program úgy válaszol, hogy megkísérli az adott elemet megtalálni vagy előállítani.

Ennek a folyamatnak az eredménye látványos. A közismert „Hanoi torony” feladat, amely az embert 5–6 korong felett már komolyan próbára teszi, Prolog-ban egy (a „program-adminisztráción” túl) kétsoros programmal megoldható. Az Einstein-feladat (amelyet névadója szerint az emberiség öt százaléka tud megoldani – és személyes tapasztalatom, hogy a megoldások többsége próbálgatással születik – mondatonként egy-egy Prolog-sorrá alakítható és a számítógép a megoldáson felül a megoldás helyességének a bizonyítását is megadja, valamint annak a bizonyítását is, hogy több megoldás nem létezik. A folyamat eredménye látványos; a módszere viszont primitív. Itt a megoldások kizárólag próbálgatás alapján alakulnak ki: a számítógép sorról sorra (abban a sorrendben, ahogyan a programba beírtuk!) megkísérli „behelyettesíteni” a tényeket a programba írt összefüggésekbe, s ha ez a lépés sikeres, akkor továbblép az összefüggés még ki nem elégitett részéhez. Ha pedig egy behelyettesítés nem sikerül: akkor a Prolog egyet visszalép, az előző rész-összefüggést megpróbálja más behelyettesítéssel kielégíteni és így tovább. Az egész folyamat a labirintusban közlekedő „gép-egér” ismert alap-algoritmusához hasonlít: ha nekimegy a falnak és az adott pontból nem tud továbbjutni, akkor visszatolat a legutóbbi olyan elágazásig, ahol még nem merített ki minden továbbhaladási lehetőséget, és tovább próbálkozik. A folyamat vége vagy az, hogy eljut a sajátig, vagy pedig hogy minden lehetséges útvonalat végigjárt.

A látszólagos ellentmondás (primitív módszerrel elért különleges eredmény) mert a gép a primitív módszert a maga elképesztő, felfoghatatlan sebességével és tévedhetetlenségével alkalmazza.

A magunk mérésében Prolog-nyelven fogalmaztuk meg a feladat megoldásához szükséges előismeret- és szabály-halmaz, s ezek alapján a program – nem kevés próbálkozási lépéssel ugyan – létrehozta a helyes megoldást. Szó nincs arról, hogy a gépi megoldási menet az emberi gondolkodást utánozza; a gép feladata az, hogy valahogyan válaszoljon a feltett kérdésre. Ebben az esetben, amikor egy számítógépi eljárás eljut egy megoldás megkonstruálásáig, többféle következtetést is sikerült megalapoznunk. Tudjuk már, hogy a szóban forgó feladat valóban feladat, nem pedig probléma: megadható, egzakt, véges előismeret- és szabály-halmaz birtokában megoldható.

Tudhatjuk, hogy a szóban forgó feladat a vizsgált személyek számára is feladat. Ehhez csak arról kell meggyőződünk, hogy a megadott, egzakt, véges előismeret- és szabály-halmazt ismerik: egy adott, egzakt, véges előismeret- és szabály-halmaz ismerete azonban megfelelő diagnosztikus teszttel egyértelműen bizonyítható. (Én magam a mérések első végrehajtásakor egyfajta könnyebb utat választottam: a vizsgált személyeknek, a feladatmegoldás előtt, kiosztottam azt a leírást, amely – persze „emberi nyelven” megfogalmazva – tételesen tartalmazta a Prolog-program által „ismert” előismereteket és szabályokat.)

A Prolog-program ugyanakkor, a megoldás megkonstruálásának folyamatában, az összes lehetséges jó megoldást bemutatja, sőt ha akarjuk: az összes lehetséges jó megoldáshoz vezető összes lehetséges gondolatmenetet is. Nem fordulhat tehát elő, hogy egy em-

beri megoldási gondolatmenetet hibásnak találjunk csak azért, mert hosszadalmasabb, vagy csak mert más, mint amire a kísérletvezető gondolt. Minden lehetséges megoldás és azok minden lehetséges útvonala előttünk áll – és a továbbiakban tevékenység-mintáztaként kereshető a vizsgált személyek gépi naplóiban.

Még nagyobb „csemege”, hogy a számítógépes eljárás feltérképezi az összes lehetséges gondolkodási zsákutcat is, mégpedig pontosan addig a pontig, ahol ezek zsákutca volta bizonyíthatóan kiderül. A későbbiekben vizsgálható lesz tehát az is, hogy hol, mikor következtek el a vizsgált személyek gondolkodási hibát, és hány, milyen lépés kellett ahhoz, hogy ezt észrevegyék. Hibára vezető döntés-helyzetek tipizálása, önellenőrzési módszerek hatékonysága – felsorolni is nehéz a felmerülő gondolatokat, amelyek további vizsgálatokra sarkallhatnak.

Amire nagyon vigyázni kell: ahogy tudomásul vesszük, hogy a gép által követett feladatmegoldási folyamat nem az emberi gondolkodást utánozza, ugyanúgy nem szabad feltételeznünk ennek a fordítottját sem. A Prolog sajátosságai és a feladat gépi megfogalmazásának módja miatt a jelen esetben a program az összes lehetséges megoldási gondolatmenetet megmutatja; de fel sem merülhet, hogy az emberek feladatmegoldó képességét aszerint osztályozzuk (a szó bármely értelmében), hogy a megoldás hosszabb vagy rövidebb, a gép által előbb vagy később megtalált, gépi szempontból egyszerűbb vagy bonyolultabb lépés-sorozat mintájára született-e.

A mintázat-illesztéses próbálgatásokon alapuló Prolog-program „jó megoldásokat” állít elő – ugyanúgy, ahogy a véletlenszerű karaktersorozatokat kinyomtató program is „írhat verset”. Az eredményen nem látszik, de mi már ismerjük, hogy miben áll az azonos eredményhez elvezető gépi és emberi folyamat közötti eltérés.

Összegezve

A fenti két eljárás közül az első elemzi, a második modellezi a feladatmegoldás folyamatát. Az elemzés eredménye az, hogy jellegzetes gondolkodási lépés-kombinációkat (próbálgatás, adott séma használata stb.) azonosíthatunk: így lehetővé válik, hogy a megoldás folyamatát annak eredményétől elvonatkoztatva elemezzük, nagy tömegben, azonos kritériumok szerint. A modellezés hozadéka a megadott előismeretek elégséges voltának, a feladat feladat voltának bizonyítása, a megoldás során szóba jöhető összes lehetséges (akár megoldáshoz, akár zsákutcháoz vezető) konzisztens gondolatmenet teljes körű bemutatása.

Mindkét eljárás a Prolog mesterséges intelligencia-nyelvet használja. Remélem, a cikk kellően meggyőző abban a tekintetben, hogy a Prolog-megoldás és az emberi megoldás egyike sem „mérhető a másiknak a mércéjével”.

Végezetül fel kell tennünk a kérdést: vajon a leírt módszer egy egyedi helyzet egyedi különlegessége-e, vagy más helyzetekre is általánosítható. Ezzel kapcsolatban azt kell megvizsgálni, hogy a cikkben említett alkalmazás miben és mennyire egyedi.

A feladat feladat voltának bizonyításakor minden előismeret, minden következtetési szabály egyértelműen megfogalmazható volt. A megoldás(ok) lehetséges készlete) véges és a megoldás helyes vagy hibás volta egyértelműen bizonyítható. Ez nemcsak a számítástechnikai feladatokban van így; más területeken a Prolog-ba való átültetés talán fáradtságosabb, de nem lehetetlen. Más kérdés, hogy esetleg olyan át-értelmezést, kódolást kíván, amely az áttekinthetőséget s ennek kapcsán a Prolog-megoldás bizonyító erejét csökkenti.

A megoldási folyamat elemzésekor viszont kihasználtam, hogy a vázolt feladatban a feladatmegoldás közben automatikusan áll elő a „gépi napló”. Más területen ezt emberi jegyzőkönyvvezéssel s annak átkódolásával kellett volna megoldani, ami megint csak az egyértelműség, áttekinthetőség, s ezzel a bizonyító erő rovására mehet.

A pontos alkalmazhatóságban s annak gyakorlati használhatóságában tehát nem lehetünk biztosak; bízom azonban abban, hogy a módszer, a felhasznált eszköz (amelynek ilyenfajta felhasználásáról a szakirodalomban nem találtam nyomokat) esetleg mások számára is megfontolásra érdemes ötleteket adhat.

Utóirat

Ennek a cikknek az alapötlete akkortól származik, amikor a sajtó a „Kaszarov kontra sakkozógép” játszma-sorozat hírével volt tele. A hír ma már nem aktuális; a tanulságai talán igen.

Remélem, az Olvasó nem tart cinikusnak, ha elmondom, hogy képtelen voltam átérezni a helyzet izgalmát. Sajnálatos, de ilyen „szenzációkhoz” még csúcstechnika sem kell: a legegyszerűbb daru is bármikor legyőzi a súlyemelés világbajnokát, egy segédmotoros kerékpár is gyorsabb és kitartóbb a maratoni futóknál, és a régi, rozoga, rácsos ajtajú liftek

Einstein híres mondása, mely szerint „a számítógép képes kérdésekre válaszolni, de nem tud önálló kérdést feltenni”, frapárs kiindulás a következő gondolathoz: fogalmazzuk meg valamiféle gépi formában a kérdéses előismereteket – s ha a gép azok alapján valóban meg tudja adni a választ, akkor ezzel kétség nélkül bizonyítottuk, hogy az előismeret- és szabályhalmaz teljes, és hogy alkalmazásukkal (minden alkotó gondolkodás nélkül) a kérdés megválaszolható: vagyis hogy a feladat valóban feladat.

is felülmúlják a rúdugrás magassági rekordját. Mellesleg biztos vagyok abban, hogy egy jól programozott mikroprocesszor már ma is jobban vezetne egy Forma 1-es autót, mint a versenyzők legjobbika. Gondoljuk csak meg: ahol nem pontosan az a cél, hogy a „puszta embert” tegyük próbára, hanem ahol az eredmény és a biztonság a fontos, ott ma is darut használunk, motorral hajtott járművet, liftet, robotpilótát. Hadd kockáztassak meg egy pimasz jóslatot: évtizednyi időn belül nem lesz kártya- vagy táblajáték, amelyben ne volnának bajnokokat legyőző kutyák és ketyerék. Ami szabályokba foglalható, azt a gép jobban, megbízhatóbban, kitartóbban, gyorsabban, olcsóbban csinálja; mi, emberek, éppen ezért készítünk gépeket, nemde?

Őszintén szólva nincsenek kisebbségi érzéseim a sakkozó embert legyőző gép miatt, mint ahogy a többi felsorolt példa miatt sem,

és amiatt sem, hogy nem tudok olyan szépen énekelni, mint amilyen szép ének a hanglemezen van – már tudniillik, ha előzőleg felvették rá valakinek a szép énekét! Ugye értjük egymást? Jobban sakkozik-e az ember a gépnél: ez a kérdés értelmetlen, mert a gép nem sakkozik! A gép algoritmusokat hajt végre, amelyeket mi a jelen esetben egy sakkjátszma lépéseiként „dekódolunk”, de a gép tevékenységének annyi köze van a sakkhöz, mint a DVD-lejátszónak a filmművészethez. A sakkozó kutyák voltaképp egy *Neumann*-féle számítógép, amely ennek révén arra és csakis arra képes, hogy konzerválja és koncentrálja ki tudja, hány programozó kódolt gondolatait a sakkjáték tényeiről és szabályairól. Ha ezeket sikeresen kombinálja és illeszti az adott játszma adott „mintázataihoz”, ezt az „emberi elme” sikereként és nem kudarcaként élem meg.

Kaszarov elveszített egy szuper-szimultánt, amikor a sakkozógépet konstruáló szakértők és programozók ezreivel mérkőzött meg egyszerre, úgy, hogy neki a sakkóra által kiszabott idő állt rendelkezésére, amazoknak pedig a fejlesztés éveit. Ez semmiben nem csökkenti az én izgalmamot és örömömet, amikor nagyfiammal sakkozom és egyszer-e egyszer a huszadik lépésig is eljutok, mielőtt tönkreverne. Ez a két dolog nem ugyanarról szól: az egyik a hegymászó örömteli erőfeszítéséről, a másik a hegyoldalba sífelvonót építő ember technikai diadaláról. Ha a kettőt össze akarjuk hasonlítani, akkor az er-

re vonatkozó klasszikus példánál vagyunk: egy alma és egy körte közül melyik a jobb cseresznye?

Remélem, hogy (egészen más indíttatású) cikkem ebből a szempontból is olyan gondolatokat kelthetett az Olvasóban, amelyek érdemesek a továbbgondolásra. Egykori kedvenc geometria-tankönyvemből vettem azt az idézetet, amellyel ezt a gondolatmenetet zárom: „Mi csak megmagyarázzuk álláspontunkat, de nem állítjuk, hogy ez az egyetlen lehetséges álláspont.” Lectori salutem!

Irodalom

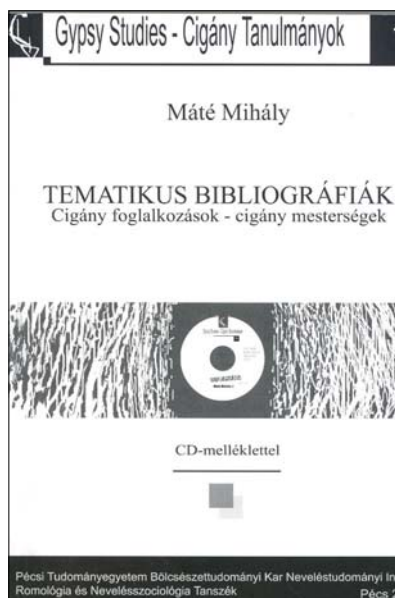
Briars, D. J. – Larkin, J. H. (1984): An integrated model of skill in solving elementary word problems. *Cognition and Instruction*, 1. 245–296.

Csikos Csaba (2003): Egy hazai matematikai felmérés eredményei nemzetközi összehasonlításban, *Iskolakultúra*, 8. 20–27.

Hajós György (1960): *Bevezetés a geometriába*. Budapest.

Mayer, R. E. – Hegarty, M.: A matematikai problémák megértésének folyamata. In: Sternberg, R. J. – Ben-Zeev, T. (szerk., 1998): *A matematikai gondolkodás természete*. Vince (Kulturtrade) Kiadó, Budapest.

Reusser, Stebler (1997): Every word problem has a solution – the social rationality of mathematical modeling in schools. *Learning and Instruction*, 4. 309–327. Idézi Csikos Csaba (2003): Matematikai szöveges feladatok megértésének problémái 10–11 éves tanulók körében. *Magyar Pedagógia*, 1. 35–55.



A PTE BTK könyveiből